



Inside This Issue

From the Publisher	1
IIGS ADB Keyboard Repairs	
by M H (Buzz) Bester	2
Daisychaining a Pair of Disk][Drives	
by Vincent Quinn	3
The Color of Webbing	
by Ryan Suenaga	6
Talking][...Aaron Pulver	
by Ryan Suenaga	9
News and Announcements	10

From the Publisher

This issue welcomes a new writer, not only to *The Apple Blossom*, but to the A2 community in general: Vincent Quinn. Hailing from Ireland, Mr. Quinn is an experienced computer user, but is new to the Apple II community. I'm sure you'll agree after reading his article that he is a most welcome and valuable addition. Also appearing in this issue are returning *Blossom* writers Buzz Bester and Ryan Suenaga (with two articles!). I almost feel like I'm being pushed out the door.

Well, I'll be heading out the door soon enough—to head toward Kansas City, Missouri, where I'll be attending my first-ever KansasFest. I'm looking forward to meeting some of the big-names in the Apple II world, and I'll give you a full report on the news that is sure to flow from this annual landmark event in the life of the A2 Community.

I'll also be making another move this August (about 2 days after I get back from KFest, in fact), when my family moves into a house we're buying. Fortunately, the business PO Box provides a stable address, and my many email addresses (stevec1021@aol.com, appleblossom@delphi.com and abpub@ziplink.net) will remain the same as well. So keep in touch!

I wrote a few months ago about trying to create themed issues, and while we didn't manage to do that with our second issue this year, we do have a pretty solid theme in this issue—hardware fixes galore. We hope to finally create the hypermedia-

centered issue I've mentioned before after KansasFest, where there may very well be a major hypermedia announcement (and it doesn't concern *Hyper Quarterly*, which has been a pretty big event in itself). But that's all you're getting from me for now!

Speaking of *Hyper Quarterly*, the new disk subscription is going well, with more than 30 subscribers so far. There's a web page about the first issue on the Apple Blossom Publishing web site. To read about the new issue and see screen shots of the stacks (downloadable with Lynx), point your browser at:

http://members.aol.com/newblossom/hq/bq_index.html

I also wrote earlier this year about the possibility of a new release of Publish It! 4. While I have hope that this will still happen, negotiations have been slow with the current copyright holder, and I don't expect this to be resolved before KansasFest. Hopefully it will be done by Christmas! There is interest in upgrading the program and I hope that we will be able to restore this lost gem in the Apple II software treasury.

Finally, I remind you to keep on with your Apple II. Not only is it just as good as it was when you first got it, but it's actually better. New software, like StolenBase, continues to be written for it, and new hardware becomes available as well. The journey is far from over.

APPLE IIGS KEYBOARD PROBLEMS

Special to The Apple Blossom by M H (Buzz) Bester

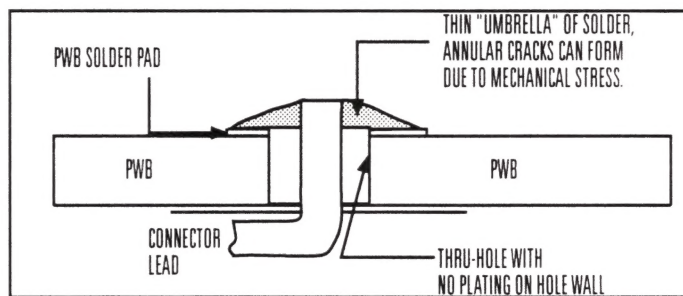
I once thought a rare-event was a comet colliding with Earth, winning the Lottery, or a IIGS keyboard connector developing a fault! Now, IIGS keyboard connector faults seem to be more common and are therefore a timely subject. Two out of the four keyboards, Apple part number (P/N) 658-4081, that I have owned needed repair, and about 20% of our Orange Apple (II) Computer Club members reported having problems over the years ranging from paper-clips and food between keys to open circuits where the Mini-DIN connector sockets solder to the Keyboard printed wiring board (PWB).

Contamination jamming the keyboard is not uncommon and requires pulling off key caps and cleaning; that procedure was well described by Glenn Ferreri, on page 20 of *II Alive* July/Aug, '93. With regard to the open-circuit solder repair, Apple IIGS (and Macintosh) keyboards have two identical, interchangeable, Mini-DIN connectors at the rear, opposite ends. Either of these can be used to connect the ADB (Apple Data Bus) cable to the CPU; the remaining connector is for the mouse. An open circuit at the ADB-to-keyboard connector manifests itself by "freezing" keystroke commands, the keyboard cursor, and the mouse cursor; the CPU is unresponsive to both keystrokes and mouse movement. However, if keystrokes function OK but the mouse-cursor is "frozen" the fault is probably at the opposite connector, where the mouse plugs in. If you suspect one of these troubles, reverse the ADB (keyboard-to-CPU) cable connector with the mouse cable connector at the keyboard (*be sure to turn off power to the computer before unplugging the ADB cables—ed.*). That swap, by the process of elimination, should identify which of the two connector sites are faulty. Sometimes it is possible to produce an intermittent open circuit at a faulty connector by stressing (bending) the cable connector while it is inserted in the Mini-DIN socket. It is highly improbable (almost a rare event!) for both the left and right end connector sites to malfunction simultaneously. If the above procedure identifies a bad connector, the problem is most likely a faulty joint where the connector is soldered to the keyboard PWB, but probing across those solder joints with an ohmmeter has not been successful in detecting which joint(s) is open.

The nature of the faults that I have found and repaired were either annular, microscopic cracks around a connector-to-PWB solder-joint, or a crack across the copper trace immediately adjacent to one of those connector solder joints. I first addressed this problem in the 1993 Nov/Dec issue of *II Alive*.

That article described the nature of those solder joints, and included a rigorous repair instruction that would pass a critical, NASA quality review. That sophisticated "remove-retin-resolder" procedure was an "overkill" and probably discouraged novice solderers. I have since found a less sophisticated repair that works as well and is not intimidating for the less dexterous among us; and if your repair attempt fails, there is consolation in that you can find used keyboards selling for \$25 to \$35.

For the courageous out there who are experiencing the keyboard and/or mouse connector malfunctions described above, reflowing the solder joints at the bad Mini-DIN socket is the recommended corrective action. Access to those solder joints is simple. Remove the ADB cables from the keyboard, flip it over, then remove the three Phillips head screws and washers from the recessed holes in the bottom panel. The long plastic strip that the screws came out of will drop off and you can then lift out the PWB to which the keys and key-switches are attached. Examine the areas where the Mini-DIN sockets are soldered to the PWB thru-holes.



Each socket has 5 solder joints; the largest joint is the ground connection and not likely to be faulty.

A two-minute lesson on soldering is appropriate at this point. Our intention is to "reflow" 4 of the 5 solder joints where the faulty MiniDIN socket is connected to PWB traces. To "reflow" solder it is only necessary to add fresh flux and remelt existing solder. Molten solder that runs down the connector lead can be replaced by adding small amounts of fresh, bright, molten solder by transferring it from the tip of the iron to the joint being reflowed. That procedure, properly done, should eliminate micro-cracks that cause intermittent or open circuits. Remelting is accomplished with a clean, freshly tinned soldering iron (not a soldering gun because guns tend to overheat rapidly). The tip of our freshly tinned iron needs to be tapered to a point, similar to a lead-pencil. It is important to avoid larger, blunt

tips because they can easily cause unintended “solder-bridges” between the closely spaced solder joints. Radio Shack’s 15 watt soldering iron, P/N 64-2051 (or equivalent) should be suitable providing the blunt tip is replaced with a pointed tip similar to P/N 64-2052. Remember that even a 15 watt iron can burn anyone who handles it improperly; also, the use of safety glasses when soldering is a good practice. You will need activated rosin flux similar to Radio Shack P/N 64-021, and small diameter, flux cored, Sn60 (60/40), or Sn63 (63/37) solder wire similar to Radio Shack P/N 64-005. The flux and solder wire are first used to “tin” (solder-coat) the soldering iron as soon as it has reached soldering temperature. An uncoated iron oxidizes rapidly at elevated temperature and is difficult to “tin” when in that condition. Use a moistened, household, cellulose sponge for wiping the tip of the soldering iron to keep it bright. Once tinned, the bright tip should be kept coated with solder to minimize deterioration by oxidation. Be sure you have a suitable holder for the hot soldering iron to avoid burns to your work area. For lack of a better holder I have used an iron skillet. Also, there is lead in solder, so adequate cleanup of residues is sensible. The iron tip will stay brighter longer if it is frequently fluxed and recoated with fresh solder. Only flux needs to be applied to an existing solder-joint being reworked. Occasionally, additional solder will be needed; use the soldering iron tip to transfer only small amounts of bright, molten, solder to the joint being reworked. Unintentional solder bridges will be avoided by careful positioning of the iron and transferring only small amounts of molten solder. A properly reworked (reflowed) solder joint will be concave and bright with no contamination or protruding points (icicles). Attempting to add solder other than transferring it with the tip of the iron can cause bridging or unwanted wicking down the PWB through-holes. Use of adequate flux avoids formation of “icicles” or points.

Note that a green insulating film of “solder-mask”

covers the copper conductors (traces) leading away from the solder joints on the PWB. It is necessary that all the solder-mask be scraped off the trace close (within about 1/16-inch) of the solder joint being reflowed. I use the tip of a very narrow screwdriver to scrape away solder-mask, but the end of a straightened paper-clip should work as well. You need to abrade away the green film until the surface of the trace is bright, bare, copper; in that condition it will wet nicely with flux and solder. Your reflowed solder needs to wet to both the bare copper surface of the trace and the tip of the Mini-DIN connector lead that protrudes through the old solder-joint. Your reflow procedure, properly done, will repair any micro-cracks that existed in the old solder-joint or across the copper trace at the solder joint. When you are satisfied that you have reflowed the possibly faulty solder joints, and done so with no solder-bridges between solder-joints or between adjacent traces, it is time to clean up flux residues. Brush your reflowed areas with isopropyl (rubbing) alcohol using an old toothbrush. Allow the alcohol to run off the PWB and blot any remaining residues with a paper towel. Examine your work with a magnifying glass, and, if your reflowed joints are clean and bright, and if no solder bridges are visible, you may reassemble your keyboard with confidence that you have done a good job!

A final comment regarding other keyboards. To satisfy my curiosity, I recently disassembled a similar but newer “APPLE KEYBOARD II, 5V 40ma” and found that the Mini-DIN connector sockets had been redesigned and were unlikely to develop the solder-joint problems found in Apple P/N 658-4081.

The author, M H (Buzz) Bester, is an engineering consultant who has spent most of his professional life on reliability improvement, cost reduction, and documentation of electromechanical components and subassemblies.

Daisy-Chaining a Pair of Disk][Drives

Special to The Apple Blossom by Vincent Quinn

Note: This article owes a great deal to the information contained in David Empson’s piece on attaching a 19-pin 5.25-inch drive to a 20-pin Disk][controller card (“Making a Disk][adapter cable”), which can be consulted on-line at:

<http://www.visi.com/~nathan/a2/faq/diskiiicable.html>

Thank you David. I’m happy to acknowledge this information debt!

Description and disclaimer:

This article describes a procedure I have used to modify three Disk][drives. Previously, all three were standard Disk][drives, attaching to a Disk][controller card in slot 6 of an Apple IIGS, via a 20-strand flat ribbon cable with IDC connectors at each end. Now they have 19-pin plugs and pass-through sockets, allowing any two of them to be daisy-

chained to each other and to one or more Apple 3.5-inch drives on the IIGS smartport, where they are recognized as Slot 6 drive 1 and Slot 6 drive 2. Any one of the drives can also be plugged into the external disk port of a IIc (the IIc will accept only one external 5.25-inch drive) where it is recognized as Slot 6 drive 2.

Although the procedure is straightforward once you have the cable pinouts (provided that you can correctly identify the pin numbers and wires of all the connectors), it is very easy to get the wiring wrong. If you cross-wire any of the pins, you risk damaging your drive, your computer or doing other serious damage. So although I have got it right each time, I cannot guarantee that you will—and therefore I will not be responsible for any such damage. If you follow this procedure, you do so entirely at your own risk.

What you will need

The tools involved are:

- Phillips head and flat blade screwdrivers
- hand or power drill and drill bits
- baby hacksaw or 'abrafile'
- fine file
- fine-nosed electrician's pliers
- soldering iron
- continuity tester

For each drive you intend to modify, you will need:

- one 19-pin male D plug
- one 19-pin female socket
- one 20-pin IDC connector
- about six inches of 20-strand flat ribbon cable, with strand 1 marked (this is usually in red)
- about 15 inches of round 20-core-and-shield cable.

The pinouts:

The ones that concern us are the drive control signals that are common to the IIGS smartport, the IIc external disk port and the Disk][pins. They are as follows (again, thanks to David Empson):

Signal	IIGS	IIc	Disk][
ground	1,2,3	1,2,3,4	1,3,5,7
-12V	5	5	9
+5V	6	6	11,12
+12V	7,8	7,8	
13,15,17,19			
write protect	10	10	20
stepper motor	11,12,13,14	11,12,13,14	2,4,6,8
write request	15	15	10
drive 1 enable	17	17	14
drive 2 enable	9	-	14
read data	18	18	16
write data	19	19	18

To complete the picture, there are some other sig-

nals which, in the case of 5.25-inch drives, we can safely ignore. They are:

Signal	IIGS	IIc	Disk][
external interrupt	-	9	-
3.5 drive enable	4	-	-
3.5 drive select	16	-	-
no connection	-	16	-

From the above, we can see straightaway that the significant signals at the IIGS smartport and the IIc external disk port are practically the same—the major difference being the lack of a drive 2 enable signal on the IIc. Note also that there is only one enable pin—pin 14—on a Disk][. We are going to use this to our advantage.

A note before we begin: the most difficult thing I found about this conversion was finding 19-pin D plugs and sockets—I couldn't! What I had to do was to get 23-pin ones and cut them down with a hack-saw (at least, I cut the plugs down—where the sockets were concerned, I simply blocked up the last hole at the left and right ends of each row with super glue). They work fine...

Phase one—the physical work

Obviously, you need to carry out all work with everything switched off and with your Disk][drive(s) disconnected from the controller card.

First you need to cut a mounting hole in the Disk][casing to take the pass-through socket. Turn the drive upside down and remove the four black phillips-head bolts. Then slide the main part of the drive forward out of the casing.

Carefully unplug the 20-pin IDC connector that attaches the ribbon cable to the drive's logicboard. (Important—note carefully how this plug and cable are attached: the ribbon cable comes out of the IDC plug on its *upper surface*.) Also note that the pin numbers of the connector are marked on the drive's logicboard—this is again very important. Detach the ribbon cable from the cable restraint on the back plate of the drive casing and put it aside—you should keep it in case you ever need to use your drive with a Disk][controller card again.

Now turn the drive over again and remove carefully the four countersunk screws that hold the mechanism itself to the bottom part of the case. Lift the mechanism gently away. If you wish, you can give it a cleaning and recalibrate its speed while you have the drive apart (instructions on adjusting speed and drive alignment can be found on-line—check the Apple II FAQ).

Decide where on the back of the casing you want to locate the pass-through port. I've found that the best place for cutting the hole is (if you look at the casing from the rear, holding it the right way up)

around 15 mm down and 15 mm in from the top left corner (peel off the red sticker that says 'Warning—connect this cable only to the Disk][controller card': I mean, you ain't gonna need *that* anymore!). If you put the socket there, there is little danger of fouling other connections inside the drive when it is reassembled.

Use paper and pencil to draw a cutting template around your 19-pin socket. In my case, to make sure that any rough edges would be concealed (and to make the job a little easier!) I mounted the flange of the socket on the outside, not the inside, of the drive casing—bear this in mind when I talk about feeding the cables through the mounting hole from inside to outside below: you don't need to do this if you are mounting the flange of the socket inside the casing. Then scribe the hole you have marked onto the back of the drive casing. Use the drill to make a series of small holes, joined together so that you can get the hacksaw or abrafile through them to begin cutting. Carefully cut around the template hole (the back casing of all three of my drives was aluminum, so this wasn't too difficult) and remove the waste; smooth any rough edges with the file.

Now bring up the socket to the hole and make sure it fits. Mark the positions of the two mounting bolt/rivet holes, take the socket away again and drill the two holes out. Now wash any metal swarf off the casing. The physical part is done!

Phase two—the electronics

First strip the outer insulation from both ends of your 20-core-and-shield cable. We are going to use only 19 of the cores, so select one core and cut it back to the outer insulation at each end. Tin (coat with solder) the other 19 cores at both ends, and tin the solder cups of both the 19-pin plug and socket.

Now solder the 19 cores to the 19-pin male D plug—it doesn't matter which color wire goes to which pin, but all 19 pins must have a wire going to them.

Pass the free end of the cable through the hole that you have cut in the drive casing, passing it from inside to outside of the casing. Now here comes one of the critical parts. Using your continuity tester, check which core you have soldered to which pin of the 19-pin plug. I'd recommend checking three times and writing the wire colors down carefully so that there is no mistake. Now fold the wires that run from plug pin 9 and plug pin 17 back out of the way, and solder the remaining 17 cores to the female 19-pin socket like to like: ie, the wire running from male plug pin 1 goes to female socket pin 1, 2 goes to 2, etc.

Now you have 19 cores attached to the plug and 17 to the socket: at the socket end, there are no wires running to pin 9 or pin 17 and you have two cores unattached. Here is another critical bit. Take

the wire that runs from male plug pin 9 and solder it to female socket pin 17. The wire that runs from male plug pin 17 remains unattached for the moment.

Identify the pins on the 20-pin IDC connector. The easiest way of doing this is as follows: hold the IDC connector horizontal, with the pin holes down and the notch that indicates the side of the connector that the cable is to come out of facing you. Pin 1 is at the left front (i.e., the one nearest to you). Using a vise and two bits of wood, clamp the 20-pin IDC connector to the six inch flat ribbon cable, taking care that the red stripe that indicates strand 1 of the cable actually runs to pin 1 of the connector. If you do this right, the strands of the cable will run to the correct pin numbers: ie, the red stripe will run to pin 1, the strand on its right will run to pin 2, the strand on *its* right to pin 3, etc. Using a piece of wire pushed into the holes of the IDC connector; use your continuity tester to make sure that everything is OK and also that there are no shorts between adjacent pins caused by faulty clamping. Again, this is one of the critical points: you must know which wire runs to which pin.

At the other end of the ribbon cable, bare and tin the ends of the wires. Now carefully split the insulation so that you can peel wires 14, 17 and 19 back and away from the rest. Cut off 17 and 19—they are not needed. Remember the wire coming from the male plug pin 17 that we left loose? Solder it to wire 14 coming from the IDC connector, cover the joint with tape or heatshrink and gently move it aside. Feed the remaining 17 wires that come from the IDC connector through the socket mounting hole, from the inside to the outside, and solder them to the pins exactly according to the following plan:

Signal	IDC 20 wire	female D socket pin
ground	1	1
stepper motor	2	11
ground	3	2
stepper motor	4	12
ground	5	3
stepper motor	6	13
ground*	7	4
stepper motor	8	14
-12V	9	5
write request	10	15
+5V	11	6
+5V**	12	16
+12V	13	7
+12V	15	8
read data	16	18
write data	18	19
write protect	20	10

*3.5-inch drive enable on the IIGS;

** 3.5 head select on IIGS; not connected on IIC.

There should be nothing connected to pin 9 of the female D socket.

That's the electronics finished!

Put it back together:

First bring the 19 pin female socket up to the mounting hole and fix it in place with two small bolts or rivets. Now attach the 20-pin IDC connector to the logicboard of the drive, taking care that the red strand goes to pin 1 as marked on the drive's board (If you have done everything correctly, the ribbon cable should come out of the IDC connector on top). Secure the mechanism to the base of the casing with the four countersunk screws. Clamp the round 20-core cable into the cable restraint on the back of the drive, then slide the mechanism backwards into the casing, feeding the cable out as you go. Secure the casing with the four black phillips-head bolts. Finally, put the cable restraint and the shells on to the male 19-pin D plug.

Now repeat this procedure exactly with a second drive, so that you have identical devices. Now the drives can be daisy-chained on a IIGS smartport: as they are identical; it does not matter which one is first or second: whichever it is, it will be identified as 'Slot 6 drive 1' or 'Slot 6 drive 2', as appropriate.

How it works

It is very simple, really. As you can see from the pinouts above, the wiring means that the signals from the IIGS or the IIC ports are passed straight to the correct pins at the IDC connector on the drive's logic board. The clever bit is where the two drive enable signals end up.

The Disk][requires its drive enable signal on pin 14. The IIGS (and the IIC) puts out a drive 1 enable signal on pin 17. As is clear from the above, the signal on pin 17 at the computer is transmitted directly to pin 14 on the Disk][, because we connected the wire from pin 17 of the 19-pin male plug directly to pin 14 of the IDC connector.

The IIGS puts out its drive 2 enable signal on pin 9. We connected pin 9 of the 19-pin male plug to pin 17 of the female pass-through socket. So when you daisy-chain two of these devices together, the second one sees the drive 2 enable signal at pin 17 of the first drive's female pass-through socket and passes it to pin 14 at the IDC connector on the logic board. Daisy-chainable Disk][s. Simple, isn't it?

By the way, if you are lucky enough to come across 19-pin plugs and sockets, would'ya mail me some?

The author, Vincent Quinn, is a recent Apple II-convert who hails from County Dublin, Ireland.

The Color of Webbing

A Review of Spectrum Internet Suite by Ryan Suenaga, M.S.W.

It is the rare soul who hasn't heard about the World Wide Web. The hype that is the Internet has gone from being strictly geekspeak to everyday conversation. *Netscape Navigator* and *Microsoft Internet Explorer (MIE)* for the Windows and Macintosh are today what *AppleWorks* and *Lotus 1-2-3* were a few years back—the very reason why you bought a computer.

While the Apple II enthusiast has always been able to access the Internet through the ubiquitous UNIX shell account, a counterpart to *Navigator* or *MIE* has been elusive. A direct connection to the Internet using the TCP/IP protocols via a PPP or SLIP dial up—commonly used with *Navigator* and *MIE*—was long believed the essential missing building block.

At various times through the last two years we have been teased by the possibility of Derek Taubert's GS/TCP project and hoped that it would evolve into the graphical Internet browser we have been waiting for. Unfortunately, waiting for GS/TCP has been like *Waiting for Godot*—and like *Godot*, the story continues on without GS/TCP.

Too Suite

From the prolific Apple II software publisher Seven Hills Software comes the *Spectrum Internet Suite*, the world's first Apple II web browser. Born in the dormitory of Avila College in Kansas City, Missouri, during KansasFest 1996, and nurtured by *Spectrum* author Ewen Wannop and Geoff Weiss, a System Administrator for a Washington, D.C.-based Internet Service Provider (ISP), the *Spectrum Internet Suite*, or SIS, is composed of a complex set of scripts, XCMDs, and displays for *Spectrum*, the blockbuster Apple IIGS telecommunication program.

A group of KFesters, including Weiss, Wannop, prodigious Australian programmer Richard Bennett, Genie and Delphi A2Pro sysop Tim Buchheim, and Seven Hills Software's Dave Hecker got together at the Apple II world's premier event in July of 1996 and pondered the possibility of a web browser without TCP/IP (for more details on this, see recent articles in Joe Kohn's *Shareware Solutions II*, Max Jones' *Juiced.GS*, or *GenieLamp A2*). The result is SIS, a program which links the future of computing—the prodigious Internet—and the past—a ten

- year old computer with roots in the 1970s. In fact, no program in recent memory carries the banner of the Apple II more truly—it is Internet without insanity, the World Wide Web without wackiness. In the best tradition of the Apple II, SIS gets the job done without the hype. SIS does not display graphics like browsers on other platforms, but it acts a lot—make that *a lot*—like graphical web browsers with graphics turned off.

SIS Shapes Up

Shortly after placing my order, the package with the SIS disk arrived in the mail—with nothing else in it. In what may come as a shock to traditionalists, Seven Hills Software has composed the SIS manual in the HyperText Markup Language and placed the documentation for SIS on the World Wide Web itself! (The same HTML files are available on the SIS disk) Purists may scoff, but doing documentation this way saves the publisher a tremendous amount of time, money, and shelf space, and that savings, at least in this case, is passed on to the end-user (witness the SIS price—a meager \$25). It also makes updates to the manual simple and straightforward, and for those of you who still insist on printed documentation, it's an easy matter to use SIS's built-in ability to translate the HTML into Teach format and print it from *ShadowWrite*, *EGOed*, or any of a host of Apple IIGS NDAs or applications.

SIS requires either a UNIX shell account with an Internet Service Provider (ISP) or a Genie account. Unfortunately, it won't work, at least as is, with the VAX system used by Delphi, PPP or SLIP only connections, or with UNIX accounts that force you to use a menu rather than direct commands.

All SIS-tems Go!

I made a backup of the non-copy protected SIS disk, read the prerequisite READ.ME files, and proceeded to install SIS. This simple setup process shows you the flexibility of *Spectrum*—the installer isn't a separate stand alone program, it's *Spectrum* itself, running an installer script.

Setting up SIS is straightforward, but you will need some information handy when you start. You'll need to specify which type of connection you'll be using (either a UNIX shell account or the Genie online service), and if you're using a UNIX shell account and want to send out electronic mail from within SIS, you need to supply your name, email address, and the name of your SMPT (email) Server. If you're lost at this point, your ISPs support crew ought to be able to give you a hand. (Please note that if you use SIS via Genie, you will not be able to use SIS's email features.)

Once SIS was installed, I dialed my local ISP and signed on (with a *Spectrum* script, of course). When I

got to the main prompt, I ran the START.SIS script with the Command-R command, and my *Spectrum* display morphed from the classic ANSI emulation to a colorful splash screen to what looked like an Apple IIGS approximation of a *Netscape Navigator* display. SIS was up and running!

At the top of the SIS screen is the familiar Apple IIGS menu bar, which you will notice is different from the one usually in *Spectrum*, but in addition to the menu bar options, there is a tool bar and an entry line reminiscent of the one used by *Navigator*.

While the *Spectrum Internet Suite* does not have all of the features of the big name web browsers on other platforms, its feature list compares favorably with the latest and greatest. For example, SIS shows off a history list of your most recently visited pages, as well as a bookmark list of your favorites. It can cache pages to disk (even a RAM disk, if you have sufficient memory) for quick retrieval and a "progress bar" to keep you apprised of page downloading status. It can even complete your Uniform Resource Locator entries automatically (eg., if you type in "sevenhills", it will send out a request for "http://www.sevenhills.com"). SIS will even handle pages with frames, the same way the latest versions of *Lynx* do.

With SIS, you can do many of the same things you did with *Lynx*, but see the results better. When you're filling out a form on the WWW, for instance, you can see a better picture of what the form's designers wanted the form to look like.

Sins of the SIS-ter

Currently, the *Spectrum Internet Suite* has some decided limitations compared to the big name browsers on other platforms, and even the utilitarian UNIX-based *Lynx* browser to which Apple II users have grown so accustomed.

While many will bemoan the lack of GIF and JPEG graphics that are displayed with browsers on other platforms, I don't see this as a great deficiency (likely because I tend to run graphical browsers on the Macintosh with graphic loading turned off, in search of speed). What I see as SIS's largest deficiency is the inability to download files. In this regard, *Lynx* has SIS beat. It's relatively easy to configure *Lynx* to download files to your local terminal with a number of different protocols. SIS, unfortunately, can't download files at all.

While SIS supports many different URL types, it does not work with the ftp://, news:, gopher://, and https://. While you can read and post news using an alternative method (such as the DejaNews WWW site at http://www.dejanews.com), no work-arounds for the other URLs have been found...yet.

Many WWW pages are designed for graphical web browsers on other platforms that have higher reso-

lution than that on the Apple IIGS. While the built-in font used with SIS was specially designed with the resolution limitations of the Apple IIGS in mind, it will be helpful to run either a larger than stock monitor on your Apple IIGS (a 17-inch monitor with a Second Sight card would be a prime choice) or a well-tuned AppleColor RGB monitor.

A hot-rodded Apple IIGS is recommended to run SIS. Four megabytes of RAM, a hard drive, an accelerator, and a 14.4 modem ought to be considered requirements for browsing the Web with SIS.

User response to SIS has been quite enthusiastic. There have been users who have used *Spectrum* since version 1.0 who have had to delete their preferences file when installing SIS—apparently, SIS doesn't like converted preference files. There are other minor quirks, like the fact that just pressing Return after entering a URL isn't enough to send SIS to the new site; you must hit the "green light" button, and the need to double-click rather than single-click a highlighted link. But these are minor quirks, barely worth mentioning.

A Decided Improvement

While SIS doesn't have the ability to download files or display GIF or JPEG graphics, it does, as you might expect, have some special features that other browsers do not have.

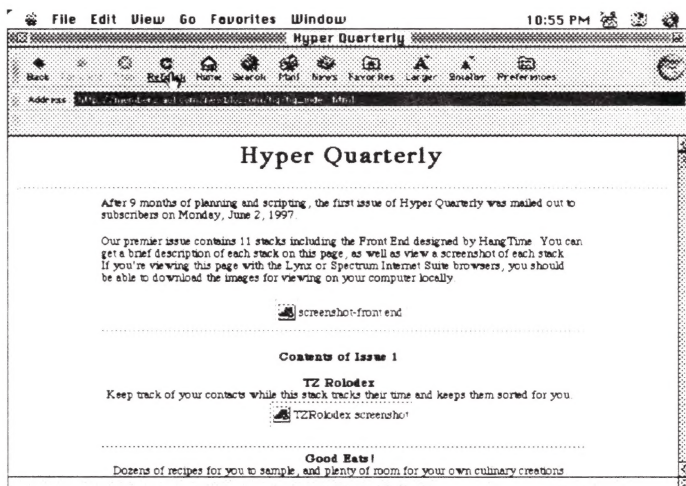
Because you're fortunate enough to use an Apple IIGS, you are in for a real treat with SIS. SIS is tuned to use some of the unique capabilities of the Apple IIGS. SIS will display some special characters when you use it with special HTML tags; for instance, the `<grin>` and `</grin>` tags will show smiling faces to match those of the Apple IIGS user cruising the 'Net with SIS. You can play sounds with your Web pages, and if you have Talking Tools from the Byte Works installed, your Web pages can be set up to talk!

Sizing Up SIS

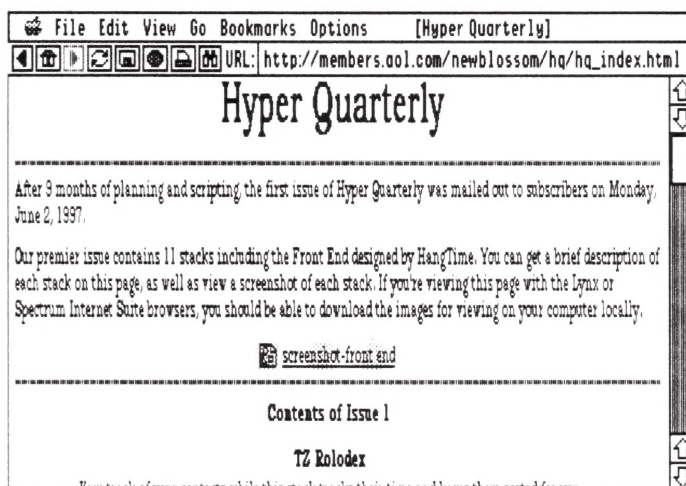
The *Spectrum Internet Suite* uses the capabilities of the Apple IIGS to its fullest without trying to do too much. It's a speedy, well thought out program from a prolific software publisher interested in supporting the Apple II world.

While the Apple II, particularly the Apple IIGS, remains a valuable and useful tool for getting work done, the platform has lacked a web browser for quite some time. While the *Spectrum Internet Suite* won't fulfill the expectations of someone who expects all of the capabilities of the latest versions of *Netscape Navigator* (like downloading files, displaying GIF or JPEG graphics and playing QuickTime movies), SIS is a no-nonsense alternative to the bloated browsers on other platforms, while adding its own unique capabilities in the best tradition of the Apple II. The *Spectrum Internet Suite* shows once

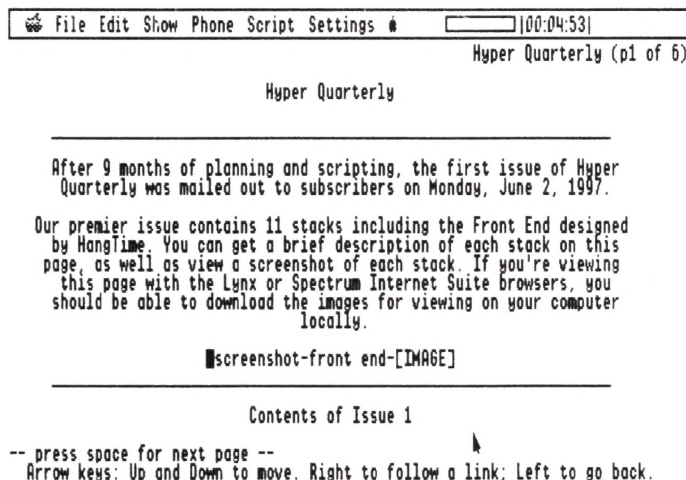
again that there is plenty of fire left in the belly of the Apple IIGS.



The Hyper Quarterly Web page viewed with MS Internet Explorer for Macintosh



The Hyper Quarterly Web page viewed with Spectrum Internet Suite on a IIGS.



The Hyper Quarterly Web page viewed with Lynx.

Talking][...Aaron Pulver

by Ryan Suenaga, M.S.W.

Aaron Pulver, one of the Real Geeks of the Apple II world, put out a very needed program earlier this year with the release of StolenBASE, the first GS/OS based MIME (a standard internet email encoding scheme) encoder/decoder. He then released that same program as "Delphiware", freeware only distributable online by the Delphi online service. We caught up with Aaron recently and talked to him about StolenBASE, Delphiware, and other things Apple II.

The Apple Blossom (TAB): When did you first start using an Apple II?

AP: I was introduced to Apple II computers at a friend's house in '84 or '85, back when I was in school. It was an Apple IIe with one 5.25-inch disk drive, and nothing else. Not even an 80-column card or printer... Still, my friend and I would sit in front of that thing for hours, mostly playing games, like One-on-One, Dig-Dug, Drol, and many other "classics". It wasn't long before I started doing a little Applesoft programming, and then I was hooked. I got my own IIe a few years later, and a year after that, I sold it and bought my ROM 01 IIGS. The rest is history...

TAB: The program StolenBase which you created and recently released has been met with much applause. What other programs have you created for the Apple II to date, and what do they do?

AP: That's a tough question. For 8-bit Apple IIs, I wrote so many little programs during my "IIe years", that I really can't remember too many of them. There were a couple of Applesoft hi-res paint programs, a hi-res character generator and character editor, a few utilities, and of course, many small games. Out of all those, the only program I released was Disk Menu, a menu-based program selector for ProDOS 8. I uploaded it to Quality Computers' BBS back when they had one, but I doubt it made it beyond there. For the IIGS, there have been quite a few programs, but once again, I've only released a few of them. That's mainly because I was missing a few reference manuals and would run into a problem that I couldn't find an answer to. One of those half-finished programs is a King James Bible CD-ROM, which I'd like to complete now that I have a complete collection of reference manuals. As for released programs, there was Helmet to Helmet (a football game done with a friend), Play CD (a CDA that plays audio CDs from ProDOS 8), MIME64, and

StolenBASE. I think that's been it so far, actually. I've only had a Delphi account since February, so I haven't had a chance to go back and upload my older programs yet.

TAB: Your StolenBase program was something that was really, really, really needed! What got you inspired to work on StolenBase?

AP: Actually, I was looking for base64 source code for a project at work, and the C code I found looked pretty simple. Of course I had to try compiling it on the IIGS, and a few changes later, it did compile. So, now I had ported a shell application that could encode and decode base64 files. It wasn't very pretty, but I released that as MIME64 on Delphi. I could see some problems with that program, though... For one, not every IIGS owner has a shell like ORCA or GNO/ME. It was also slow and not very user-friendly, so of course I just had to do something about that. At first, I had just planned on sticking the existing code into a desktop program, but after thinking about it for a while, I decided to start from scratch and do the main work in assembly language. Well, it took a bit longer to complete, but I was pretty happy with the results. A couple of bug fixes and features later, StolenBASE is at version 1.1. So really, the whole thing was just to see if I could do it.

TAB: Do you have any plans for new Apple II programs in the future, and if so, what are they?

AP: Of course I have plans. StolenBASE v2.0 is on the way, for one. It will be an NDA, not to mention the additional features... I have two games that may be out later this summer, and then there are a couple of bigger projects waiting for my time, too. I'll be keeping busy.

TAB: What's your favorite computer language and why?

AP: Hmmm... I really can't say I have a favorite. I'm kind of a computer language geek, you know. On the GS, I probably use Pascal the most, and assembly a close second. I use C some, but not as much as those two. I still use Applesoft, and Microl BASIC GS, but not generally for serious projects. At work, I use a bit of Java, C++, and Visual BASIC, but the main one is Perl. Nice variety, isn't it?

What kind of advice would you give the beginning programmer?

You can never have too many reference manuals and you should program constantly. Books are for picking up a new trick or two, and practice keeps your skills sharp. Maybe that's not the best advice, but that's how I've done it.

TAB: There seems to be a little controversy regarding your releasing StolenBase as "Delphiware". Can you explain what Delphiware is, and your reasons for releasing software like that?

AP: Delphiware is just freeware that is only available online from Delphi. If others want to use that label on their software, they can go right ahead. It was really just something I thought up when I was about to release StolenBASE. I didn't really know how well StolenBASE would work, since I hadn't tested it much, so I wanted to keep the release limited to Delphi. Then, the more I thought about it, the better I liked the idea. There aren't a huge number of Delphi Apple II users yet, so it's kind of a "reward" to those who are. And if it brings in new members, that's great! I do plan on releasing most of my programs as Delphiware. As for StolenBASE, I'll eventually release to the public either a decode-only version, or v1.1 once I have v2.0 out. My view is that telnet access to Delphi is really cheap, so people should be able to get an account if they want Delphiware.

TAB: What do you think about the A2 effort on Delphi?

AP: It's great! I never could afford Genie, and comp.sys.apple2 just doesn't have the same friendly atmosphere. Ahem. Anyway, I've made some new friends and met a lot of new people on Delphi. So, having an active Apple II group on Delphi is great for me, and for many others, I hope.

TAB: What do you do for a living? Does it involve computers as well?

AP: Of course it involves computers... I'm not a geek for nothing. Seriously, I work for NorthLand Computer Exchange, Inc. in Bemidji, Minnesota. Development and support of email gateways is our main business, but there is plenty of variety. It is a full-time job, though, and I do mean full. It doesn't leave all that much time for working on the IIGS during the week, so you can bet I'm up late on week-ends.

TAB: If you could be remembered for just one thing by the Apple II Community, what would you want it to be?

AP: While I've been involved with Apple II computers for a long time now, I've only been involved with the Apple II Community for a short time. I guess I still have to find my place, in many ways. How would I like to be remembered? Ask me that in ten years. I'm not going anywhere.

News and Announcements

Genie RoundTable Closings

The A2Pro RoundTable (RT) on Genie has merged with the A2 RT, due to the RoundTable not pulling in enough members. A number of other RTs, Syndicomm and non, also closed at the end of June.

Many New Offerings from SSII

With Vitesse and EGO Systems leaving the Apple II market, Shareware Solutions II is picking up several of their products to keep them available for the Apple II community.

The Quickie handscanner software is currently at version 3.2, and is available to subscribers of *Shareware Solutions II* for \$5.00. A Quickie interface card is required for the software's operation, so this will be of interest only if you have the scanner but not the latest software, which has many updates.

The *Harmonie* GS/OS printer drivers are now published and sold exclusively by Shareware Solutions

II. The price has just been lowered to \$20, and that includes shipping to anywhere in the world. Shareware Solutions II is now accepting orders for *Harmonie*, complete with manual. New printer drivers are also expected to be on the disk, including an updated StyleWriter driver.

SSII is also distributing the Westcode software products formerly distributed by EGO Systems, including *Pointless* (to use True Type fonts on the IIGS) for \$20.00, *TypeWest* (a collection of 40 professionally designed True Type fonts) for \$15.00 and *TypeSet* (a utility that displays font faces in the font pull-down menus of GS/OS programs along with cataloging utilities to produce sample printouts of your font collection) for \$15.00. SSII is also distributing *HardPressed*, a disk compression software that compresses files on your hard drive transparently so that you continue to work with the files normally but gain disk space because of the compression.

- *HardPressed* is also \$20.00. Finally, *InWords* is Optical Character Recognition (OCR) software that works with the Quickie or LightningScan hand-held scanners to turn a scan into editable text. *InWords* sells for \$20.00.

SSII has also picked up EGO Systems' *Addressed for Success* database/label program. *Addressed for Success* allows for many different label designs, and can import AppleWorks 3.0 and 4.0 databases, as well as tab-delimited text files. AFS is available for \$20.00, or in a bundle with *ContactsGS* for \$35.00.

Joe Kohn
Shareware Solutions II
166 Alpine St
San Rafael, CA 94901

News from Seven Hills Software

Seven Hills Software is offering a special bundle price for those who would like to get both *Spectrum 2.1* and *SIS*. *Spectrum* is normally \$85 and *SIS* is normally \$25, but both are available at a special bundle price of just \$95, plus \$3.50 shipping and handling in North America (outside North America it's exact shipping costs but at least \$3.50 per order).

Seven Hills is pleased to announce that anyone with the capability to get on the Internet can access their new website at www.sevenhills.com/applesoftware. (And that capability is available to GS users, now that *SIS* is here!) Besides lots of GS software, they have a few Mac products and a bargain-bin closeout of Mac and PC software from other companies.

Seven Hills' new tech support address is support@sevenhills.com. Use this, from any online service you use, whenever you have a technical question about any of their products. Note that any technical questions sent to another email address of theirs, or given to them by phone will just be forwarded to the support@sevenhills.com address and that will delay the answer you get. But, please send any other correspondence to one of their other addresses (such as shss@ix.netcom.com). The old email address at nettally.com is no longer functioning.

For those who are waiting for the next version of *SuperConvert*, the company says that they think they're on their last beta copy and the update to that program will be the next new product they ship. If you don't currently own *SuperConvert*, but would like to purchase it when it's at version 4.0, then please email Seven Hills and tell them you'd like to be on their "notify-when-available" list for *SuperConvert 4*.

It's safe to say that it is going to be at least another two months (perhaps we'll see it at this year's KansasFest?) before the new version is ready. In the meantime, you can look forward to the huge increase in conversion filters that will be in

SuperConvert 4 (about 400% more than the current 3.01 version).

In late Spring, Seven Hills experienced an unprecedented failure of the hardware and software they use for their ordering system. Consequently, they know some names and orders have been lost, but are not yet sure which those are. They believe they will be able to fill all orders that were placed before April 25. If your order hasn't arrived, email (preferred) or call them. Send email about this to shss@ix.netcom.com.

Also, check your credit card statement to make sure you were not charged twice. In the process of trying to get orders out "by hand" they might have unintentionally made this mistake. They will immediately credit your card upon receipt of a faxed or mailed photocopy that indicates you have been overcharged.

Seven Hills has reduced prices for their GS software. For more information, if you already have the means to go online with *SIS* or another browser, you can find information at the following address:

<http://www.sevenhills.com/applesoftware/iigs/sis/index.htm>

If you can't go to their website then email them for more information about their GS products and new prices. You can use their autoresponder by simply putting gsinfo@sevenhills.com as the email address and leaving the body of your message blank.

Seven Hills Software
1254 Ocala Rd.
Tallahassee, FL 32304-1548
Phone: 904-575-0566
Fax: 904-575-2015
General email: shss@ix.netcom.com
Technical support email: support@sevenhills.com

Apple CEO Resigns under Pressure

Apple Computer, Inc.'s CEO and Chairman of the Board Dr. Gilbert Amelio resigned his positions on July 9, 1997. Speculation is that the Board of Directors was unhappy with Apple's continued fiscal problems. While the company has improved its financial position, and has released several new products this year, including a fast new PowerBook and is readying a new release for Macintosh System Software (Mac OS 8), it has also continued to lose money and has failed to get its message across to the buying public.

Related to this is an expanded role at Apple for Steve Jobs, who is on the search committee for a new CEO as well as providing increased attention to Apple as an advisor to the Board of Directors. While Jobs has never been an especially close friend of the Apple II, his return to Apple could bring back some

of the pioneering spirit that impelled the company to produce its first computers.

Alltech Electronics Lowers HardCard Prices

Alltech Electronics sells the Focus HardCards which we reviewed last year. The hard card is actually an IDE hard drive mounted on an interface card that plugs into one of the expansion slots of your Apple II+, IIe, IIGS or Laser 128 computer. According to Alltech's Web site, prices have been lowered again. Here's the info:

Focus HardCard	Price
20 MB	\$ 59.00
60 MB	\$ 99.00
80 MB	\$119.00
120 MB	\$149.00
170 MB	\$169.00
200 MB	\$199.00

Alltech Electronics
2618 Temple Heights
Oceanside, CA 92056
phone: (619) 724-2404
fax: (619) 724-8808

SPACS Report

A while back we reported on an email project at the Stevens Point Area Catholic Schools that would include Apple II computers in the local area network. I recently heard back from Jack Friess who told me about some very exciting developments in the project such as:

- 1) Creating a menu system for Linux (a freeware UNIX running on PCs) for A2 users.
- 2) Discovering Kermit-65 freeware to use as an A2 terminal client and:
 - Writing (by Laine Houghton) a shell menu system for Kermit-65 called Miss Piggy.
 - Reworking the Kermit startup routines to boot right to the network

The Apple Blossom
c/o Steve Cavanaugh
P.O. Box 120434
Boston, MA 02112-0434

- 3) Writing (by Jay Schulist) a Linux driver for the PC LocalTalk card to bring Macs cheaply onto the network through LocalTalk and AppleTalk.
- 4) Automating many of the network processes.
- 5) Creating (and continuing to create) set-up routines to easily install and configure Linux for the Mac/PC/A2 network.

Mr. Friess says that, "It will be criminal for schools to spend \$50,000 to \$100,000 or more per LAN when they could spend \$5,000 to \$10,000 or less per site. We will be finalizing the documentation and 'How To' manual this Fall...so any school can 'do it themselves'."

The SPACS networking project has a new URL, which continues to grow along with the project.

[http://www.spacs.k12.wi.us/spacs/technology/projects/low\\$net/a2email/NETdocs/docindex.html](http://www.spacs.k12.wi.us/spacs/technology/projects/low$net/a2email/NETdocs/docindex.html)

The Byte Works Puts More Books Back in Print

Joe Kohn reported in his latest *Shareware Solutions II* that Mike Westerfield of The Byte Works has acquired the rights to put several important IIGS programming books back in print. These books were formerly published by Addison-Wesley and have been out of print for some time. The books and prices are:

Book	Price
IIGS Toolbox Reference 1	\$45.00
IIGS Toolbox Reference 2	\$45.00
IIGS Toolbox Reference 3	\$55.00
IIGS GS/OS Reference	\$30.00
IIGS Firmware Reference	\$30.00
IIGS Hardware Reference	\$30.00

The Byte Works
8000 Wagon Mound Dr., NW
Albuquerque, NM 87120
phone: (505) 898-8183

